An Approach to Identify Indic Languages using Text Classification and Natural Language Processing

Deepthi Shetty
Computer Science & Engineering Department
Nitte Meenakshi Institute of Technology
Bangalore, India
deepthijeevan27@gmail.com

Sarojadevi H
Computer Science & Engineering Department
Nitte Meenakshi Institute of Technology
Bangalore, India
hsarojadevi@gmail.com

Uzma Shakeel
Computer Science & Engineering Department
Nitte Meenakshi Institute of Technology
Bangalore, India
uzma.shakeel89@gmail.com

Sanjana S
Computer Science & Engineering Department
Nitte Meenakshi Institute of Technology
Bangalore, India
sanjana64885@gmail.com

Aishwarya G M
Computer Science & Engineering Department
Nitte Meenakshi Institute of Technology.
Bangalore, India
gmaishwarya6@gmail.com

Nupur P
Computer Science & Engineering Department
Nitte Meenakshi Institute of Technology
Bangalore, India
nupur3051@gmail.com

2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon) | 978-1-6654-9790-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/MysuruCon55714.2022.9972371

***Abstract*- India is one of the most culturally and linguistically diverse nations in the world. India stands second in the world for the most languages spoken by its diverse population, who speak their own regional languages for communication. English is offered as a second additional official language in India. However, there is a communication gap in India because of how little English is used there. It's nearly impossible for humans to bridge this breach by translating from one language into another. However, it is possible to translate languages by taking the help of a machine. As per the literature survey, it was observed that Neural Machine Translation (NMT) is a cutting-edge strategy that significantly outperformed more conventional machine translation methods for translating one language into another.**

The main objective of this proposed work is to achieve accurate identification of Indic language texts and scripts and provide relevant names of the language after the detection process. The entire work is carried out in stages which includes, collection of the dataset from different sources, preprocessing with the help of data mining techniques, identifying the language of input and in future, approaches like rule based, statistical and neural networks will be used followed by post-processing and efficient tasks like Machine Translations, Named Entity recognition, etc. will be carried out.

Keywords - Language detection, Neural machine translation, statistical machine translation, text classification, vectorization.

I. INTRODUCTION

India is the most diverse country in the world in terms of language where it can be said that India has become a land of many tongues and has been called “as a tower of veritable languages”. In 1950, the States in India were reorganized on a linguistic basis. The Constitution of India has approved 22 languages. There are 121 official languages and 270 mother tongues combined. 96.71% of the population in India speaks one of the scheduled languages as their native tongue, while 3.29% speak other languages. This creates a communication barrier between English and the regional languages of India.

The Constitution of India has approved 22 languages. There are 121 official languages and 270 mother tongues combined. 96.71% of the population in India speaks one of the scheduled languages as their native tongue, while 3.29% speak other languages. This creates a communication barrier between English and the regional languages of India.

Natural language processing provides an optimal translation output by enabling users to put out content in the language of their choice and allows the content consumers on the platform to consume content in their preferred language. In Natural Language Processing (NLP) computational linguistics, rule-based modeling of human language, machine learning, and deep learning are all combined to provide computers the ability to comprehend human language, which can be represented in text or audio data. The concept of NLP has been around for more than 50 years, and with the spread of computers around the world, it is currently a technical industry that is fast expanding.

The shortcomings of conventional machine translation systems are addressed by the revolutionary approach known as neural machine translation (NMT), which makes use of an artificial neural network (ANN) to create a model for the language. The majority of the source and target information and mapping storage issues that were primarily related to earlier statistical machine translation (SMT) systems are resolved by NMT. NMT has the majority of its components automated and integrated, in contrast to SMT, where each part of the language and translation models must be adjusted independently. These days, a sophisticated multilingual MT system is available. A statistical machine translation system was previously available (SMT). Neural machine translation is a rapidly developing field.

Machine Translation is evaluated on the BLEU (Bilingual Evaluation Understudy) score. BLEU is a metric for automatically evaluating machine-translated text. The score is between 0-1, the higher the score the better the machine translation. There are three major approaches: Rule-based Machine Translation (RBMT), Statistical Machine Translation (SMT), Neural Machine Translation (NMT).

978-1-6654-9790-9/22/$31.00 ©2022 IEEE

II. RELATED WORK

In the paper [1] it was observed that machine translation methods are fast and effective for processing. But are not always accurate although performance can be improved using the Seq2Seq model. The effectiveness of translation systems is influenced by a number of factors, including vocabulary size, model tuning, and linguistic characteristics of the chosen languages, according to research using neural machine translation models on Indic languages with limited resources. This was observed in paper [2], which describes experiments with neural machine translation models on Indic languages with low resources. Some of the Intelligent Approaches for Natural Language Processing for Indic Languages include attention-based models, which consistently outperformed the Long Short-Term Memory (LSTM) Seq2Seq model and Attention-based seq2Seq grasped the overall meaning of the language [3].

The segmentation might perform even better if linguistic rules and phonetic conversion were included. Prior to this, there had never been a dedicated implementation dealing with the impact of rare words in the Out of Vocabularies. The primary experimentation in automatic speech recognition focused on sub-words, which performed better than other methods and approaches at the time [4].

Lower f-measure, recall, and precision are produced by code mix scripts. This is explained by the potential for multiple faults in various processes. Romanized Indian words often have inconsistent spellings and transliteration problems. For instance, it's usual practice to shorten Hindi to the letter "h" when writing it. SentiWordNet searches for it but never finds anything, making sentiments difficult to discern. The prevalence of numerous similar patterns in linguistic writing may be a significant factor in the poorer precision. Creating a corpus of normalizations for these situations might be an intriguing strategy to take, but doing so runs the risk of overfitting and normalizing terms that don't actually require it [5].

A multi-modal machine translation system, an image guided machine translation system, and assistive technology were all used in this work. The programmed produced Malayalam captions by identifying dominant features in both text and photos [6].

The sentiment Analysis technique is suitable for checking word duplication. The ontology technique is suitable for Translating a text from one language to another. Compression results show that most techniques received various degrees of accuracy, hence, helping them with their respective findings. NLP techniques should be selected depending on the area which researchers are working on [7]. The system elaborates on translation by using an LSTM encoder-decoder architecture that is based on deep learning. Because LSTM can predict future values based on prior learning data, sequential data offers greater accuracy. The technology obtains an accuracy of between 80 and 85 percent when translating between Indian languages [8] The most recent research focuses on proverb mapping for meaning retrieval or proverb expansion. Future generations can learn from the nation's cultural, social, economic, and intellectual heritage thanks to the digital proverbs archive [9].

A WSD approach that can be used to enhance Machine Translation idea mapping as a tool for MT. This technology is affordable and simple to use [10]. Also, a rule-based model follows the rules mentioned to categorize the language it is known as rule-based machine translation system, it also uses a dictionary-based MT [11].

Complex and compound phrases are notoriously difficult to translate for machine translation systems to process because of the amount of syntactical information they include. This problem consequently has an impact on translation quality as a whole. It is safe to say that training translation systems exclusively with basic sentences would improve the quality of the translated text. However, a sizable and high-quality parallel corpus containing two natural languages is necessary for training a translation system. While there are many parallel corpora for different language pairs, simple sentence-only lexicons for low-resource languages are uncommon. In such a case, the primary goal of the current effort is to create such a parallel lexicon. To construct the same, complicated and/or compound sentences from the corpus as a whole would need to be separated out and then simplified. The work comprises Bengali and English, therefore multiple algorithms to achieve the same are described in this study. Sentences are broken up into two or more segments when complex and compound sentences are reduced to basic examples; these segments must then be aligned to make them semantically equivalent [12].

The objective of this work was to enhance language-specific encoders and decoders' zero-shot translation capabilities. To that purpose, parameter sharing approaches, Transformer layers, and selective cross-attention between the interlingua and decoders were presented as an interlingua to language-specific encoders-decoders [13].

The model was integrated with an Attention mechanism that increased accuracy in order to get around the drawback of recent NMT models. A parallel English-Gujarati corpus with 65,000 sentences yielded an average BLEU score of 59.73 on the training corpus and 40.33 on the test corpus [14].

The paper [15] suggests a comprehensive framework for neural machine translation for translating Indian languages into English, which would improve the accuracy of translation for Indian languages. The paper also provides different translation metrics that can be used to assess the effectiveness of the developed model, as well as sources from which data can be collected.

During sequence-to-sequence learning, the importance of the attention mechanism in overcoming long-term dependencies associated with the vanilla LSTM model was noted in paper [16].

A Sanskrit-Hindi and Sanskrit-Gujarati bilingual dataset development was seen due to the complexity and breadth of Sanskrit grammar as well as the lack of a parallel corpus that included Sanskrit as one of the language pairings.

Because LSTM can forecast future values based on historical learning data, sequential data offers enhanced accuracy [17].

When compared to conventional methods, the MVET-based approach plays a substantially larger role. It provides 10% better translation performance compared to Hybrid, 28% better efficiency compared to Rule based, and 38% better efficiency compared to Stat-based approaches [18].

The hybrid-based method gets the best level of accuracy by combining numerous methods into a single system. A system is suggested that translates Gujarati into English using a hybrid approach made up of RMT, SMT, and EBMT [19].

For heterogeneous NLP models, NLP-Fast is a brand-new system method that attempts to provide speedy and scalable performance. Multiple NLP models can be optimized on any hardware platform, including CPU, GPU, and FPGA, using a technique that consists of two general-purpose optimizations and one device-specific optimization.[20] Different NLP models' performance-critical procedures and related performance problems are found.
Going through these and several more relevant papers implementation using the approaches which are the best fit to the required project have been planned.

III. PROPOSED METHOD

In the overall Design of the implemented system, it is intended to first tokenize raw data, replacing sensitive data items with their substitute values, or tokens, before going through several preprocessing stages and detecting and processing individual words. To deal with problems that could occur in natural language processing, two primary processing algorithms are planned to be used:
Rule-based systems: Rule-based systems depend on specifically engineered grammatical rules that need to be created by experts in the field of linguistics, or knowledge engineering experts. This was the foremost approach to crafting NLP algorithms, and it is still used to date. Neural based and statistical approaches are equally significant as they play a significant role in contributing to proceed to the next phase of processing.

Post Processing procedures: These procedures implement various trimming processes, rule filtering, and even knowledge integration is frequently included in post processing procedures. These processes all serve as a sort. Parallelization principle is simply a way to distribute the high dimension and memory data into multiple machines so that faster training and computation of the data can be achieved. Machine learning as well NER and other crucial methodologies can be used for evaluation.
Fig 1 shows the process involved in the identification and translation process.
In the implemented system, initially the user has to give an input containing text in any language after which the text is analyzed using the text stored in the dataset using various techniques that are implemented, the language is detected and the result obtained is the name of the detected language of the entered text. In the implemented system, a few necessary modules have been imported which are also known as packages for example the python package “sklearn.model_selection” is imported so that the test, split, and train function can be used. This function splits arrays or matrices into random subsets to train and test the data, respectively. The “sklearn.feature_extraction ” module is also imported where it is used to extract features in a format supported by machine learning algorithms from datasets consisting of formats such as text and image.
The tasks that have been carried out have been explained in detail below.

- Extracting a raw text.

Initially raw text is extracted from any source like Kaggle, the dataset contains a total of 17 languages and the number of sentences in that particular language include English (1385), French (1014), Spanish (819), Portuguese (739), Italia (698), Russia (692), Swedish (676), Malayalam (594), Dutch (546), Arabic (536), Turkish (474), German (470), Tamil (469), Danish (428), Kannada (369), Greek (365), Hindi (63). Therefore, a total of 10,634 entries in the dataset. The dataset used in our implemented system has already been Pre-processed.

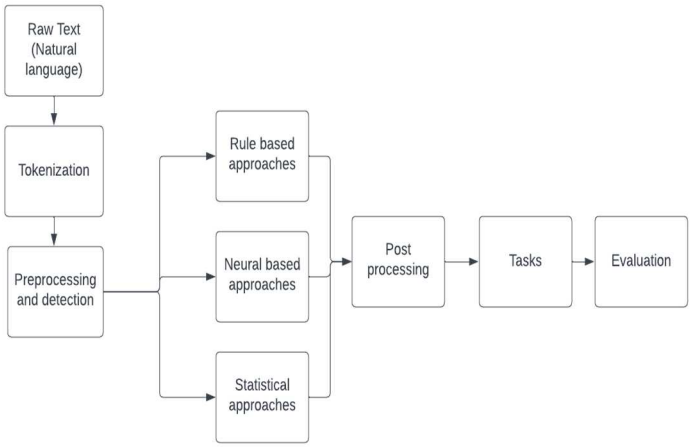

Fig 1. Design of the translation process

- Tokenization

Tokenization is a process by which the data elements are replaced by their substitute values, or tokens. Tokenization is a form of breaking down the text into smaller more manageable elements.
The “re” module is imported to use the “sub” function. This function is used to clean the data set by returning all the matching data and replacing the matched data with the desired string. Here, special characters like “!@#$(),n"%^*?:;~” are replaced with empty space(“.”).

- Language detection

With the help of the previous mentioned methods, a cleaned well- defined dataset is obtained, it is easier to detect the language, which is done using python as it has several inbuilt libraries for simplification like pandas, matplotlib, NumPy, seaborn, LinearSVC and much more. Specifically, pandas Data Frames make it simpler to comprehend data identification. With Text Extensions for Pandas, we may utilize Pandas Data Frames to represent and work with the intricate data structures used by current natural language processing (NLP) software.
One or more NLP tasks are intended to carry out in the future

- Rule based approaches

Lexical analysis deals with words individually within a text. The smallest component of a word, the morpheme, is sought for. Lexical analysis determines how these morphemes are related and changes the word into its root form. A lexical analyzer also determines the world's potential Part-Of-Speech (POS). It takes the language's dictionary into consideration.
A fragment of text that has been scanned is checked for proper structure using syntax analysis. To assess proper grammar at the sentence level, it attempts to parse the sentence.

- Statistical approaches

This approach includes extraction of features and Machine learning algorithms.

- Neural based approaches

Although more advanced, neural network approaches are fundamentally a continuation of empirical methods with parameter fitting. They entail evaluating intricate systemic interrelationships quantitatively.

- Post processing

The condensed model after applying the three types of approaches, which will be in more concise format which can further be used for effective translation of text in future.

- Tasks

These tasks include machine translation and Named Entity Recognition. Machine translation is used to translate text from one language to another without the need for a human translator. The whole meaning of the text in the original language is communicated in the target language using modern machine translation, which goes beyond mere word-to-word translation.

An earlier task in natural language processing is named entity recognition. As a component of information extraction, named entity recognition recognizes and categorizes proper nouns into preset groups, such as person, place, organization, time, and date. One of the most significant and recurring tasks in data preprocessing is named entity recognition (NER). It entails locating important information in the text and classifying it into a number of predetermined categories. With further research and analysis implementation of more tasks which will produce an effective result of accurate translation have been planned.

- Evaluation

Process of assessment, to judge the quality of the model and accuracy of translation of the mentioned Indic languages to English and vice-versa.

IV. RESULTS AND DISCUSSION

In the implemented system, initially the user has to give an input containing text in any language after which the text is analyzed using the text stored in the dataset using various techniques like pipelining, feature extraction, vectorization and null value identification that are implemented, the language is detected and the result obtained is the name of the detected language of the entered text.

In the existing technologies, the input text is translated in a literal manner where the translated sentence might or might not be grammatically correct but in the proposed system, the languages are identified and also the grammatically correct sentence can be given as the output.

The accuracy is calculated using the python function where the number of elements that are correctly recognized and the total number of elements are passed and the accuracy is calculated. Fig 2 is a graphical representation of the number of texts in each language as x-axis depicts the volume of texts present and y-axis depicts the languages present in the pre-processed dataset.

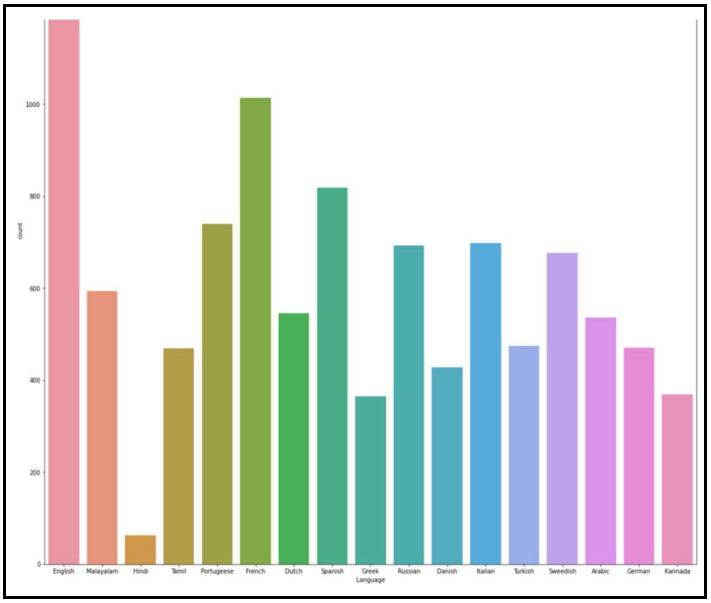

Fig 2. The bar graph shows the number of texts in each language present in the data set.

```
[ ]  print(accuracy_score(y_test, Predictions))

     0.9671746776084408
```

Fig 3. Accuracy of detection

The plaintext consisting of the foreign language is entered in the input field by the user. This system is implemented and coded using the python programming language. Google Collab is the platform that has been utilized for the implementation of the code. A few significant Python packages are employed, such as pandas, NumPy, and seaborn, to assist in the language detection process. A package called Seaborn uses Matplotlib as its foundation to plot graphs. The Sklearn train test split function aids in the creation of our training and test sets of data. It will be used to display random distributions. This is so because the original dataset often serves as both the training data and the test data. Starting with a single dataset and splitting it into two datasets, the train and test datasets by doing this, the data for a model is gathered. The Compute confusion matrix is imported, using the sklearn.metrics package. Using pandas, a data frame is made to compare real and anticipated values. Fig 3 depicts the accuracy of the language detection model which is 96%.

The detection of the language is being done on a preprocessed dataset. The accuracy study of actual language to the predicted one can be clearly seen in Fig 4 which has 2 fields, the first field contains the text in a particular language and the second field contains the name of the language in which the text has been given.

Actual	Predicted
Russian	Russian
Italian	Italian
English	English
Russian	Russian
English	English
Danish	Danish
English	English
Malayalam	Malayalam
Russian	Russian
Kannada	Kannada
Sweedish	Sweedish
English	English
Turkish	Turkish
English	English
Spanish	Spanish
German	German
English	English
English	English
English	English
Portugeese	Portugeese
Turkish	French
Portugeese	Portugeese

Fig 4. Predicted values and expected values

```
[ ] lang_clf.predict(["नमस्ते आप कैसे हैं"])

    array(['Hindi'], dtype=object)

[ ] #Tamil
    lang_clf.predict(["இன்று ஸ்ருதியின் பிறந்தநாள்"])

    array(['Tamil'], dtype=object)

[ ] # Malayalam
    lang_clf.predict(["നല്ല കാലാവസ്ഥ"])

    array(['Malayalam'], dtype=object)

[ ] lang_clf.predict(["aujourd'hui c'est l'anniversaire de shrutis"])

    array(['French'], dtype=object)

[ ] #Kannada
    lang_clf.predict(["ಇದು ಚಿಕ್ಕ ಹುಡುಗಿ"])

    array(['Kannada'], dtype=object)

    lang_clf.predict(["ಅದು ಸೇಬು"])

    array(['Kannada'], dtype=object)

[ ] lang_clf.predict(["the weather is beautiful"])

    array(['English'], dtype=object)
```

Fig 5. Language detection

The initial 25% implementation of detecting the languages, especially Indic ones is done successfully. Output is displayed in Fig 5. The results of the system implemented can be seen in the above figure.

The Confusion Matrix in the Fig 6 depicts the contrast between the test set and the predicted values and it provides an accuracy of 96% of the system. Confusion Matrix helps us analyze what the classification model is getting right and what types of errors it is making.

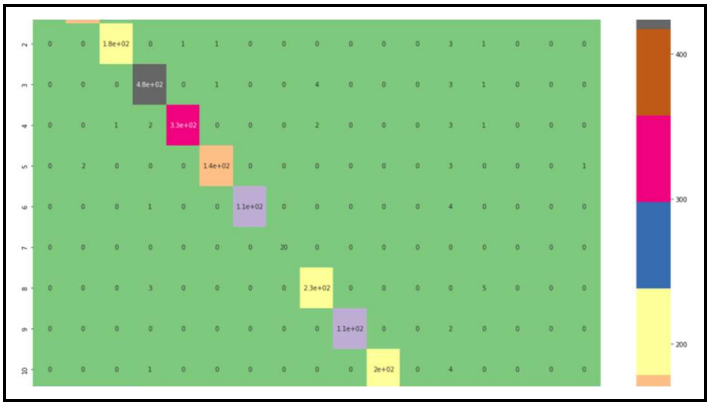

Fig 6. Heatmap of Confusion matrix showing the accuracy of the test set vs the predicted values

V. CONCLUSION & FUTURE SCOPE

The performance of translation systems is influenced by a variety of variables, including the size of the corpus, model tuning, linguistic characteristics of the chosen languages, and vocabulary size. Transformer-based models utilize the parallelization principle and include a tiered attention mechanism. In NLP, it's crucial to extract the context from lengthy statements. Long-term reliance is an issue with simple recurrent neural networks. An LSTM-based model can, to some extent, solve this issue. SMT performs better than NMT in some scenarios as it is a more efficient use of human and data resources and they are generally not customized to any specific pair of languages and have a higher accuracy rate for smaller sentences. Without taking into consideration the current translation technology, the best quality translations are always produced by customizing machine translation for a particular purpose and domain. Keeping the mentioned enumeration in mind, for future research which is planned will be able to translate a given text in English and vice versa.

REFERENCES

[1]. N. Jayanthi, A. Lakshmi, C. S. K. Raju and B. Swathi, "Dual Translation of International and Indian Regional Language using Recent Machine Translation," 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), 2020, pp. 682-686, doi: 10.1109/ICISS49785.2020.9316016.

[2]. N. Bansal, G. Datta and A. Singh, "Experimentation with NMT models on low resource Indic languages," 2021 Sixth International Conference on Image Information Processing (ICIIP), 2021, pp. 1-4, doi: 10.1109/ICIIP53038.2021.9702577.

[3]. R. Kumar and V. Sahula, "Intelligent Approaches for Natural Language Processing for Indic Languages," 2021 IEEE International Symposium on Smart Electronic Systems (iSES), 2021, pp. 331-334, doi: 10.1109/iSES52644.2021.00084.

[4]. S. Manghat, S. Manghat and T. Schultz, "Hybrid sub-word segmentation for handling long tail in morphologically rich low resource languages," ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2022, pp. 6122-6126, doi: 10.1109/ICASSP43922.2022.9746652.

[5]. R. Bhargava, Y. Sharma and S. Sharma, "Sentiment analysis for mixed script Indic sentences," 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2016, pp. 524-529, doi: 10.1109/ICACCI.2016.7732099.

[6]. L. H O and S. Jayaraman, "English -Malayalam Vision aid with Multi Modal Machine Learning Technologies," 2022 6th International Conference on Intelligent Computing and Control Systems (ICICCS), 2022, pp. 1469-1476, doi: 10.1109/ICICCS53718.2022.9788187.

[7]. Maulud, D. H., Ameen, S. Y., Omar, N., Kak, S. F., Rashid, Z. N., Yasin, H. M., Ibrahim, I. 49 M., Salih, A. A., Salim, N. O. M., & Ahmed, D. M. (2021). Review on Natural Language Processing Based on Different Techniques. Asian Journal of Research in Computer Science, 10(1), 1-17.

[8]. A. H. Patil, S. S. Patil, S. M. Patil and T. P. Nagarhalli, "Real Time Machine Translation System between Indian Languages," 2022 6th International Conference on Trends in Electronics and Informatics (ICOEI), 2022, pp. 1778-1783, doi: 10.1109/ICOEI53556.2022.9777103.

[9]. M. V. Reddy, M. H. Savant, V. N. Reddy, M. H. Savant, S. Kulkarni and N. T. Rudrappa, "Using Machine Learning Algorithm for Proverb Retrieval and Expansion for Indian Languages," 2022 9th International Conference on Computing for Sustainable Global Development (INDIACom), 2022, pp. 244-248, doi: 10.23919/INDIACom54597.2022.9763289.

[10]. P. Gupta and B. K. Joshi, "Natural Language Processing based Refining Hindi to English Machine Translation," 2022 International Conference on Applied Artificial Intelligence and Computing (ICAAC), 2022, pp. 849-854, doi: 10.1109/ICAAIC53929.2022.9792969.

[11]. R. Vyas, K. Joshi, H. Sutar and T. P. Nagarhalli, "Real Time Machine Translation System for English to Indian language," 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), 2020, pp.838-842, doi:10.1109/ICACCS48705.2020.9074265.

[12]. M. M. Rahman, M. F. Kabir and M. N. Huda, "A Corpus Based N-gram Hybrid Approach of Bengali to English Machine Translation," 2018 21st International Conference of Computer and Information Technology (ICCIT), 2018, pp. 1-6, doi: 10.1109/ICCITECHN.2018.8631938.

[13]. J. Liao, Y. Shi, M. Gong, L. Shou, H. Qu and M. Zeng, "Improving Zero-shot Neural Machine Translation on Language-specific Encoders- Decoders," 2021 International Joint Conference on Neural Networks (IJCNN), 2021, pp. 1-8, doi: 10.1109/IJCNN52387.2021.9534401.

[14]. P. Shah and V. Bakrola, "Neural Machine Translation System of Indic Languages - An Attention based Approach," 2019 Second International Conference on Advanced Computational and Communication Paradigms (ICACCP), 2019, pp. 1-5, doi: 10.1109/ICACCP.2019.8882969.

[15]. T. P. Nagarhalli, V. Vaze and N. K. Rana, "A Novel Framework for Neural Machine Translation of Indian-English Languages," 2020 International Conference on Inventive Computation Technologies (ICICT), 2020, pp. 676-682, doi: 10.1109/ICICT48043.2020.9112513.

[16]. V. Bakarola and J. Nasriwala, "Attention based Neural Machine Translation with Sequence to Sequence Learning on Low Resourced Indic Languages," 2021 2nd International Conference on Advances in Computing, Communication, Embedded and Secure Systems (ACCESS), 2021, pp. 178-182, doi: 10.1109/ACCESS51619.2021.9563317.

[17]. A. H. Patil, S. S. Patil, S. M. Patil and T. P. Nagarhalli, "Real Time Machine Translation System between Indian Languages," 2022 6th International Conference on Trends in Electronics and Informatics (ICOEI), 2022, pp. 1778-1783, doi: 10.1109/ICOEI53556.2022.9777103.

[18]. D. Patil, S. B. Chaudhari and S. Shinde, "Novel Technique for Script Translation using NLP: Performance Evaluation," 2021 International Conference on Emerging Smart Computing and Informatics (ESCI), 2021, pp. 728-732, doi: 10.1109/ESCI50559.2021.9396969.

[19]. V. A. Gandhi, V. B. Gandhi, D. V. Gala and P. Tawde, "A Study of Machine Translation Approaches for Gujarati to English Translation," 2021 Smart Technologies, Communication and Robotics (STCR), 2021, pp. 1-5, doi: 10.1109/STCR51658.2021.9588859.

[20]. J. Kim, S. Hur, E. Lee, S. Lee and J. Kim, "NLP-Fast: A Fast, Scalable, and Flexible System to Accelerate Large-Scale Heterogeneous NLP Models," 2021 30th International Conference on Parallel Architectures and Compilation Techniques (PACT), 2021, pp. 75-89, doi: 10.1109/PACT52795.2021.00013.